CLAIMS

What is claimed is:

1. An application program interface (API) embodied on one or more computer readable media, comprising:

a first group of services related to integrating content repositories into virtual content repositories (VCRs);

a second group of services related to manipulating information in VCRs;

a third group of services related to searching VCRs;

a forth group of services related to configuring VCRs; and

wherein the application program interface is compatible with a content repository service provider interface (SPI).

2. The application program interface of claim 1 wherein:

the SPI provides a subset of the services available in the API.

3. The application program interface of claim 1 wherein the first group of services comprises:

first functions to enable authorizing access to content repositories; and

second functions to enable incorporating content repositories into a hierarchical namespace; and

third functions to enable extending a VCR content model to represent information in content repositories.

4. The application program interface of claim 3 wherein:

authorizing access to content repositories includes providing authentication information to repositories and receiving authentication results from content repositories.

5. The application program interface of claim 3 wherein:

authorizing access to content repositories utilizes Java Authentication and Authorization Service.

6. The application program interface of claim 3 wherein:

incorporating content repositories into a hierarchical namespace includes representing content repositories as nodes under a single VCR root node.

7. The application program interface of claim 3 wherein:

extending a VCR content model to represent information in content repositories includes sharing a common representation of content between the API and the SPI.

8. The application program interface of claim 1, wherein the second group of services comprises:

first functions that enable creation of information in VCRs;

second functions that enable reading of information from VCRs;

third functions that enable updating of information in VCRs;

fourth functions that enable deleting of information in VCRs;

wherein information in VCRs maps to information in one or more content repositories; and

wherein information can be contents and/or schemas.

9. The application program interface of claim 1, wherein the third group of services comprises:

first functions that enable searching content information in VCRs;

second functions that enable searching schema information in VCRs; and

third functions that enable configuring search result caches.

10. The application program interface of claim 9 wherein:

searching content information in VCRs includes searching content repositories.

11. The application program interface of claim 9 wherein:

searching schema information in VCRs includes searching content repositories.

12. The application program interface of claim 9 wherein:

configuring search result caches includes at least one of: 1) setting the time to live for cache entries; and 2) setting the maximum number of cache entries.

13. The application program interface of claim 1, wherein the fourth group of services comprises:

first functions that enable configuring repository caches; and

second functions that enable configuring authorization information for content repositories.

14. The application program interface of claim 13 wherein:

configuring repository caches includes at least one of: 1) turning a cache on or off; 2) setting the maximum number of entries for a cache; and 3) setting the time to live for cache items.

15. The application program interface of claim 13 wherein:

configuring authorization information for content repositories includes at least one of: 1) setting a password and user name for a repository; and 2) setting a read-only attribute for a repository.

16. A network software architecture comprising the API as recited in claim 1.

17. A system comprising:

means for providing a first group of services relating to integrating content repositories into a virtual content repository (VCR);

means for providing a second group of services related to manipulating information in the VCR;

means for providing a third group of services related to searching the VCR; and

means for providing a forth group of services related to configuring the VCR.

18. A software architecture for a distributed computing system, comprising:

a first application configured to handle requests provided to it by a second application over a network;

an application program interface (API) to provide functions used by the first application to access a virtual content repository (VCR); and

wherein the VCR integrates a plurality of content repositories.

19. The software architecture as recited in claim 18 wherein the API comprises:

a first group of services related to integrating content repositories into virtual content repositories (VCRs);

a second group of services related to manipulating information VCRs;

a third group of services related to searching VCRs;

a forth group of services related to configuring VCRs; and

wherein the API is compatible with a content repository service provider interface (SPI).

20. The software architecture of claim 19 wherein:

the SPI provides a subset of the services available in the API.

21. The software architecture of claim 19 wherein the first group of services comprises:

first functions to enable authorizing access to content repositories; and

second functions to enable incorporating content repositories into a hierarchical namespace; and

third functions to enable extending a VCR content model to represent information in content repositories.

22. The software architecture of claim 21 wherein:

authorizing access to content repositories includes providing authentication information to repositories and receiving authentication results from content repositories.

23. The software architecture of claim 21 wherein:

authorizing access to content repositories utilizes Java Authentication and Authorization Service.

24. The software architecture of claim 21 wherein:

incorporating content repositories into a hierarchical namespace includes representing content repositories as nodes under a single VCR root node.

25. The application program interface of claim 21 wherein:

extending a VCR content model to represent information in content repositories includes sharing a common representation of content between the API and the SPI.

26. The software architecture of claim 19 wherein the second group of services comprises:

first functions that enable creation of information in VCRs;

second functions that enable reading of information from VCRs;

third functions that enable updating of information in VCRs;

fourth functions that enable deleting of information in VCRs;

wherein information in VCRs maps to information in one or more content repositories; and

wherein information can be contents and/or schemas.

27. The software architecture of claim 19 wherein the third group of services comprises:

first functions that enable searching content information in VCRs;

second functions that enable searching schema information in VCRs; and

third functions that enable configuring search result caches.

28. The software architecture of claim 27 wherein:

searching content information in VCRs includes searching content repositories.

29. The software architecture of claim 27 wherein:

searching schema information in VCRs includes searching content repositories.

30. The software architecture of claim 27 wherein:

configuring search result caches includes at least one of: 1) setting the time to live for cache entries; and 2) setting the maximum number of cache entries.

31. The software architecture of claim 19, wherein the fourth group of services comprises:

first functions that enable configuring repository caches; and

second functions that enable configuring authorization information for content repositories.

32. The software architecture of claim 31 wherein:

configuring repository caches includes at least one of: 1) turning a cache on or off; 2) setting the maximum number of entries for a cache; and 3) setting the time to live for cache items.

33. The software architecture of claim 31 wherein:

configuring authorization information for content repositories includes at least one of: 1) setting a password and user name for a repository; and 2) setting a read-only attribute for a repository.

34. A method for providing a virtual content repository (VCR) representing at least one content repository, comprising:

providing an application program interface (API);

providing a service provider interface (SPI) to be implemented by the at least one content repository; and

wherein the API and the SPI are compatible and share a common content model and a common namespace.

35. The method of claim 34 wherein the content model includes:

a set of hierarchically related objects.

36. The method of claim 34 wherein

the namespace makes addressable the content in the at least one content repository.

37. The method of claim 34 wherein the API includes:

services for performing operations on the namespace and the content model.

38. The method of claim 34 wherein the SPI includes:

services for merging contents of the at least one content repository into the namespace and the content model.

39. The method of claim 34 wherein the API includes:

a first group of services related to integrating the at least one content repository into the VCR;

a second group of services related to manipulating information VCRs;

a third group of services related to searching VCRs;

a forth group of services related to configuring VCRs; and

wherein the application program interface is compatible with a content repository service provider interface.

40. The method of claim 39 wherein:

the content repository service provider interface provides a subset of the services available in the application program interface.

41. The method of claim 39 wherein the first group of services comprises:

first functions that enable authorizing access to content repositories; and

second functions that enable incorporating content repositories into a hierarchical namespace; and

third functions that enable extending a VCR content model to represent information in content repositories.

42. The method of claim 41 wherein:

authorizing access to content repositories includes providing authentication information to repositories and receiving authentication results from content repositories.

43. The method of claim 41 wherein:

authorizing access to content repositories utilizes Java Authentication and Authorization Service.

44. The method of claim 41 wherein:

incorporating content repositories into a hierarchical namespace includes representing content repositories as nodes under a single VCR root node.

45. The method of claim 41 wherein:

extending a VCR content model to represent information in content repositories includes sharing a common representation of content between the application program interface and the service provider interface.

46. The method of claim 39 wherein the second group of services comprises:

first functions that enable creation of information in VCRs;

second functions that enable reading of information from VCRs;

third functions that enable updating of information in VCRs;

fourth functions that enable deleting of information in VCRs;

wherein information in VCRs maps to information in one or more content repositories; and

wherein information can be contents and/or schemas.

47. The method of claim 39 wherein the third group of services comprises:

first functions that enable searching content information in VCRs;

second functions that enable searching schema information in VCRs; and

third functions that enable configuring search result caches.

48. The method of claim 47 wherein:

searching content information in VCRs includes searching content repositories.

49. The method of claim 47 wherein:

searching schema information in VCRs includes searching content repositories.

50. The method of claim 47 wherein:

configuring search result caches includes at least one of: 1) setting the time to live for cache entries; and 2) setting the maximum number of cache entries.

51. The method of claim 39 wherein the fourth group of services comprises:

first functions that enable configuring repository caches; and

second functions that enable configuring authorization information for content repositories.

52. The method of claim 51 wherein:

configuring repository caches includes at least one of: 1) turning a cache on or off; 2) setting the maximum number of entries for a cache; and 3) setting the time to live for cache items.

53. The method of claim 51 wherein:

configuring authorization information for content repositories includes at least one of: 1) setting a password and user name for a repository; and 2) setting a read-only attribute for a repository.

54. A machine readable medium having instructions stored thereon that when executed by a processor cause a system to:

provide an application program interface (API);

provide a service provider interface (SPI) to be implemented by the at least one content repository; and

wherein the API and the SPI are compatible and share a common content model and a common namespace.

55. The machine readable medium of claim 54 wherein the content model includes:

a set of hierarchically related objects.

56. The machine readable medium of claim 54 wherein:

the namespace makes addressable the content in the at least one content repository.

57. The machine readable medium of claim 54 wherein the API includes:

services for performing operations on the namespace and the content model.

58. The machine readable medium of claim 54 wherein the SPI includes:

services for merging contents of the at least one content repository into the namespace and the content model.

59. The machine readable medium of claim 54 wherein the API includes:

a first group of services related to integrating the at least one content repository into the VCR;

a second group of services related to manipulating information VCRs;

a third group of services related to searching VCRs;

a forth group of services related to configuring VCRs; and

wherein the application program interface is compatible with a content repository service provider interface.

60. The machine readable medium of claim 59 wherein:

the content repository service provider interface provides a subset of the services available in the application program interface.

61. The machine readable medium of claim 59 wherein the first group of services comprises:

first functions that enable authorizing access to content repositories; and

second functions that enable incorporating content repositories into a hierarchical namespace; and

third functions that enable extending a VCR content model to represent information in content repositories.

62. The machine readable medium of claim 61 wherein:

authorizing access to content repositories includes providing authentication information to repositories and receiving authentication results from content repositories.

63. The machine readable medium of claim 61 wherein:

authorizing access to content repositories utilizes Java Authentication and Authorization Service.

64. The machine readable medium of claim 61 wherein:

incorporating content repositories into a hierarchical namespace includes representing content repositories as nodes under a single VCR root node.

65. The machine readable medium of claim 61 wherein:

extending a VCR content model to represent information in content repositories includes sharing a common representation of content between the application program interface and the service provider interface.

66. The machine readable medium of claim 59 wherein the second group of services comprises:

first functions that enable creation of information in VCRs;

second functions that enable reading of information from VCRs;

third functions that enable updating of information in VCRs;

fourth functions that enable deleting of information in VCRs;

wherein information in VCRs maps to information in one or more content repositories; and

wherein information can be contents and/or schemas.

67. The machine readable medium of claim 59 wherein the third group of services comprises:

first functions that enable searching content information in VCRs;

second functions that enable searching schema information in VCRs; and

third functions that enable configuring search result caches.

68. The machine readable medium of claim 67 wherein:

searching content information in VCRs includes searching content repositories.

69. The machine readable medium of claim 67 wherein:

searching schema information in VCRs includes searching content repositories.

70. The machine readable medium of claim 67 wherein:

configuring search result caches includes at least one of: 1) setting the time to live for cache entries; and 2) setting the maximum number of cache entries.

71. The machine readable medium of claim 59 wherein the fourth group of services comprises:

first functions that enable configuring repository caches; and

second functions that enable configuring authorization information for content repositories.

72. The machine readable medium of claim 71 wherein:

configuring repository caches includes at least one of: 1) turning a cache on or off; 2) setting the maximum number of entries for a cache; and 3) setting the time to live for cache items.

73. The machine readable medium of claim 71 wherein:

configuring authorization information for content repositories includes at least one of: 1) setting a password and user name for a repository; and 2) setting a read-only attribute for a repository.

74. A computer data signal embodied in a transmission medium, comprising:

a code segment including instructions to provide an application program interface (API);

a code segment including instructions to provide a service provider interface (SPI) to be implemented by the at least one content repository; and

wherein the API and the SPI are compatible and share a common content model and a common namespace.